Bihl + Wiedemann

Automatisierungstechnik

AS-i Master with RS232C, RS485, or RS422

User Manual

Table of Contents

1 The Used Symbols 6

2 Safety 7

2.1 Intended Use 7

2.2 General Safety Information 7

3 Connectors and Displays 8

3.1 Power Supply Concepts and AS-i Connection Techniques 9

3.1.1 Single Masters in IP65 9

3.1.2 Single Masters in IP20 with power supply A 9

3.1.3 Single Masters in IP20 with power supply N 10

3.1.4 Double Master in IP20 with Power Supply A 13

3.1.5 Double Master in IP20 with Power Supply N 15

3.2 The Serial Interfaces 19

3.2.1 AS-i Master with RS232C 19

3.2.2 AS-i Master with RS485 19

3.2.3 AS-i Master with RS422 20

3.3 Display and Operating Elements 20

3.3.1 LEDs of the Single Masters 20

3.3.2 LEDs of the Double Masters 21

4 Operating the Master 22

4.1 Peculiarities with Double Masters 22

4.2 Master Start-Up 22

4.3 Configuration Mode 23

4.4 Protected Operating Mode 23

4.4.1 Switching to Protected Operating Mode 23

4.4.2 Configuration Errors in Protected Operating Mode 24

4.5 Assigning an AS-i Address in Configuration Mode 24

4.5.1 Assigning a Slave Address 24

4.5.2 Erasing the Slave Address 24

4.6 Programming the Address in Case of Configuration Errors 25

4.6.1 Automatic Address Programming 25

4.6.2 Manual Address Programming 25

4.7 Changing the Address of Bus-Capable Masters 25

4.8 Error Messages 26

5 Operation via the Serial Interface 27

5.1 Configuring the Interface 27

5.2 Message Structure 27

6 Advanced Diagnostics for AS-i Masters 29

6.1 List of Corrupted AS-i Slaves (*LCS*) 29

6.2 Error Counter: Counter of corrupted data telegrams 29

6.3 Off-line Phase on Configuration Errors 29

7 Accessories for Putting AS-i into Operation and Test Tools 31

7.1 Windows Software AS-i Control Tools 31

7.2 DOS-Software 34

7.2.1 Frame Program ASISHELL 34

7.2.2 AS-i Monitor Program ASI_MON 34

7.2.3 Tokenizer/Loader ASI_CTRL 36

7.2.4 Visualizer ASI_VIS 38

8 Including the AS-i Masters in Own Programs 39

8.1 Telegrams of the Serial Communication 39

8.1.1 Message Structure 39

8.1.2 Message Descriptions 41

8.1.3 Representation of Information in the User Data Bytes 45

8.2 PC Drivers 49

8.2.1 New Data Types 49

8.2.2 Global Static Variables 50

8.2.3 Functions 50

9 Appendix A Seven-Segment Display 53

10 Appendix B: First Commissioning of AS-i 54

Declaration of Conformity

according to EN 45014:1989

Bihl+Wiedemann GmbH, Mannheim, hereby declares under its sole responsibility that the products mentioned below are according to the listed harmonized standards or normative documents and (where necessary) a competent body has been released.

Specification: AS-i Master / AS-i Control
with RS232C, RS485 oder RS422

Artikel-Nr.: 1030, 1031, 1032, 1033, 1034, 1035, 1042, 1043, 1044, 1045, 1046, 1047, 1105, 1135, 1136, 1137, 1138, 1139, 1140, 1147, 1148, 1149, 1150, 1151, 1152, 1105

Applied harmonized Standards:

EN 50081-2

EN 50082-2

Applied national standards or normative documents:

AS-i Complete Specification

Prüfungsordnung für AS-i Master

Manufacturer: Bihl+Wiedemann GmbH

Date: 29.01.99 Bernhard Wiedemann

1 The Used Symbols

This symbol warns the user of possible danger. Failure to heed this warning can lead to personal injury or death and/or damage to equipment.

This symbol warns the user of a possible failure. Failure to heed this warning can lead to total failure of the equipment or any other connected equipment.

This symbol gives the user important hints.

2 Safety

2.1 Intended Use

Warning

The protection of operating personnel and the system against possible danger is not guaranteed if the control interface unit is not operated in accordance with its intended use.

The device may only be operated by appropriately qualified personnel in accordance with this operating manual.

2.2 General Safety Information

Warning

Safety and correct functioning of the device cannot be guaranteed if any operation other than that described in this operation manual is performed.

The connecting of the equipment and any maintenance work to be carried out with voltage applied to the equipment must only be performed by appropriately qualified electrotechnical personnel.

In the case that a failure cannot be repaired, the device must be taken out of operation and kept from inadvertently put back into operation.

Repair work is to be carried out by the manufacturer only. Additions or modifications to the equipment are not allowed and void the warranty.

Note

The operator is responsible for the observance of local safety standards.

3 Connectors and Displays

Device in IP20:

On the front panel of the AS-i Masters with RS232C, RS485 und RS422 in IP20 are:

- terminals to connect the power supply and the AS-i circuit
- a 9-pin SUB-D connector as interface connector (RS323 / RS485 / RS422)
- 7 LEDs
- a two-digit seven-segment display
- 2 push-buttons to configure the gateway

Device in IP65:

The devices in IP65 have the terminals for the serial interface (cage clamp terminals) and the push buttons under the top of the housing to prevent liquids from entering. The AS-interface is connected with the electromechanical interface with penetration technique developed for AS-i at the bottom of the housing.

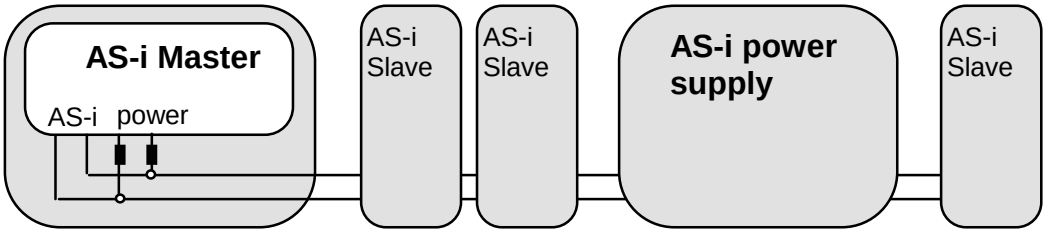
3.1 Power Supply Concepts and AS-i Connection Techniques

Note

The AS-i Masters s as well as all the AS-i gateways of the Bihl+Wiedemann GmbH are available with two different power supply concepts, "master power supply A" and "master power supply N".

3.1.1 Single Masters in IP65

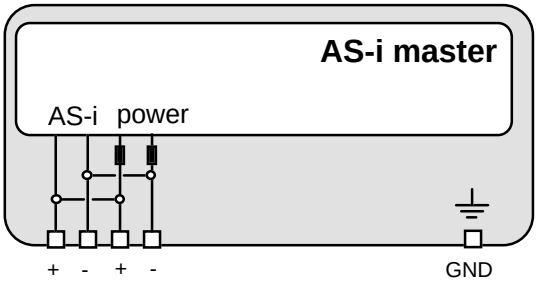
Every AS-i Masters and all the AS-i Gateways in IP65 come with power supply A. An AS-i power supply has to be used which also supplies the AS-i Masters with power and can be connected to the AS-i cable like all the other AS-i components at any place. The AS-i Masters is connected to AS-i using the penetration technique of the electromechanical interface which is common with AS-i slaves as well.



3.1.2 Single Masters in IP20 with power supply A

Note

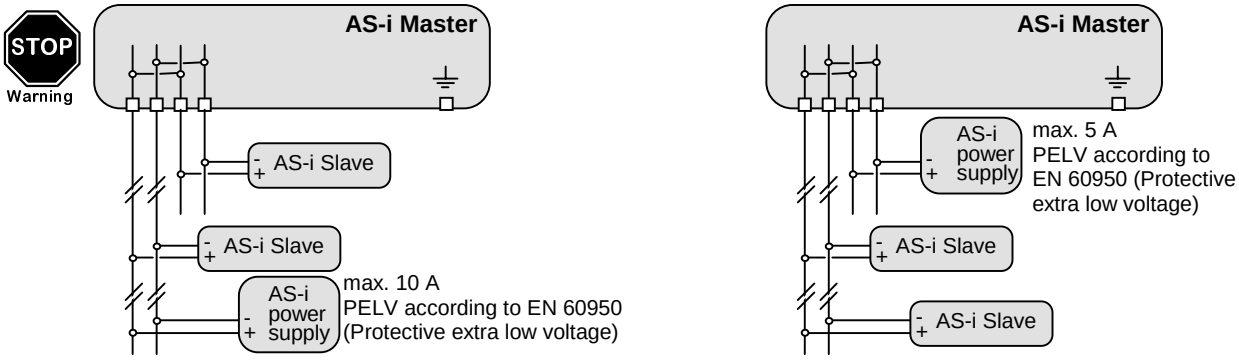
The AS-i Masters with master power supply A do not need a voltage supply of their own. They can be powered completely out of the AS-i line (the power consumption is about 200 mA from AS-i). An additional 24 V voltage source is not necessary. The AS-i Master merely requires the connection to the AS-i line. When the AS-i power supply is switched on, the master starts to operate.



The terminals have the following functions:

+	"AS-i +", Actuator Sensor Interface, positive terminal
-	"AS-i -", Actuator Sensor Interface, negative terminal
GND	Ground terminal

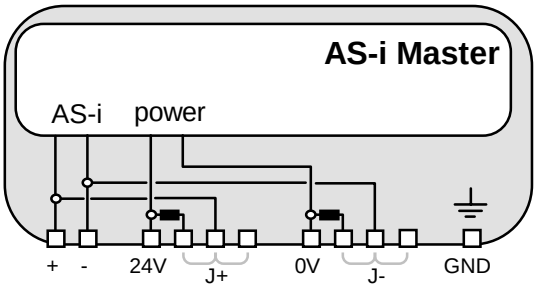
Connection samples for the AS-i power supply:



Attention **In the wiring schemes above the current through the AS-i Master must not exceed 5 A.**

3.1.3 Single Masters in IP20 with power supply N

At the bottom of the unit there are 10 terminals for the power supply and the AS-i network. The jumper settings (J+, J-) decide whether AS-i is powered by a separate AS-i power supply or by the power supply that also provides the AS-i Masters.



The terminals have the following functions:

+	"AS-i +", Actuator Sensor Interface, positive terminal This terminal is connected internally with the center contact of jumper "J+".
-	"AS-i -", Actuator Sensor Interface, negative terminal This terminal is connected internally with the center contact of jumper "J-".
24V	Master power supply, positive terminal (18 - 31.6 V DC)
0V	Master power supply, negative terminal
GND	Ground terminal
J+, J-	Jumpers for selecting the power supply of AS-i

"int pwr" position (closed):
AS-i is powered by the master power supply.
Master power supply and AS-i network are then decoupled with coils.

"ext pwr" position (open):

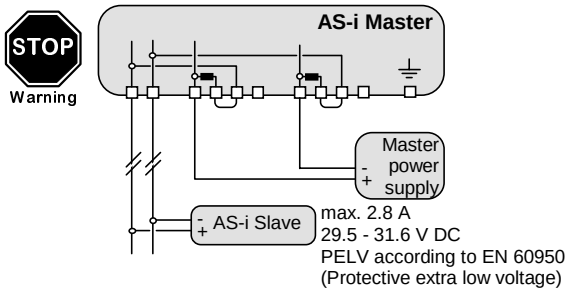
With the jumper open (or missing), the AS-i network must be powered by a separate AS-i power supply.
The AS-i power supply can be connected to the jumper or to the AS-i cable in the field.

Operation without AS-i power supply

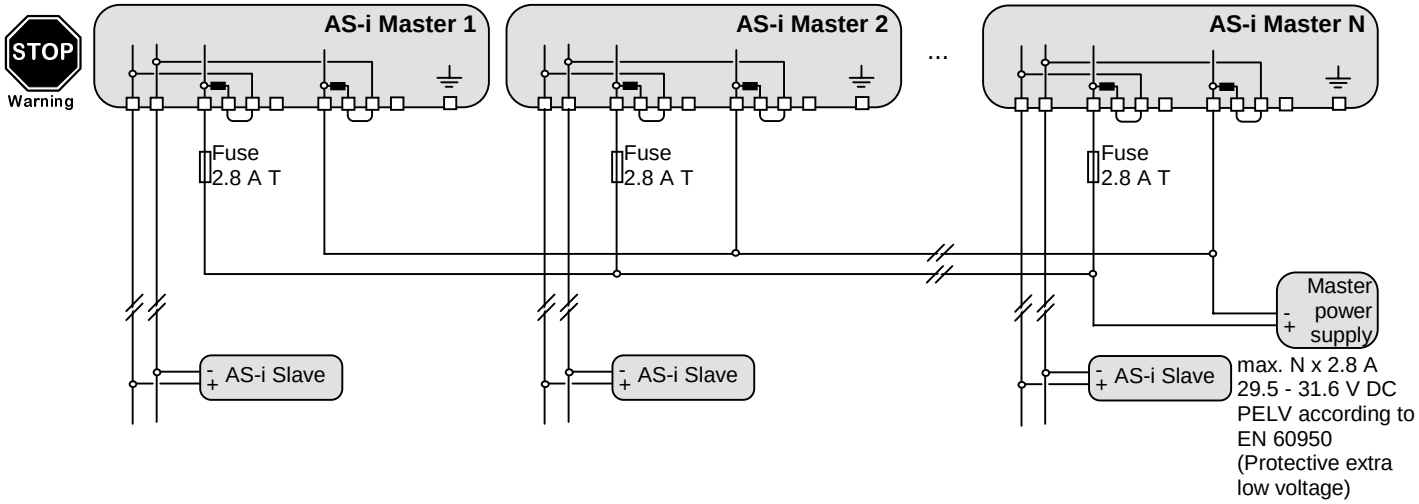
When the jumpers are in position "int pwr", the AS-i network is powered by the master's power supply. For test purposes, it is possible to use a conventional 24 V DC power supply to supply the AS-i network. You get optimum results with a 30 V DC voltage source.

Attention

For the following wiring scheme the current through the AS-i Master must not exceed 2.8 A.

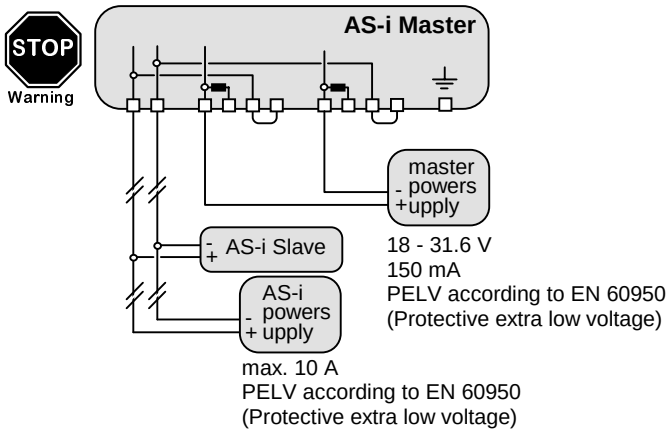


This connection technique serves also for providing more than one AS-i Masters out of one master power supply because each AS-i circuit is decoupled with its own in the AS-i Master integrated coils.

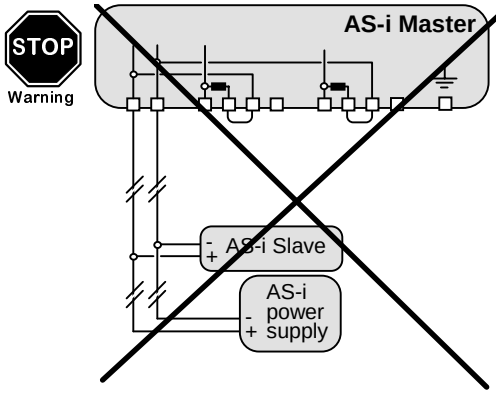


Operation with AS-i power supply

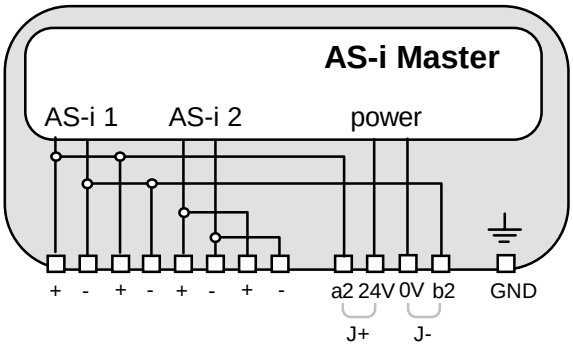
In the jumper position "ext pwr" the AS-i network needs an additional AS-i power supply.



Do not try to provide the AS-i Master with power supply N out of the AS-i power supply of the AS-i circuit („backwards“), because the additional decoupling coils would burden the AS-i line and cause erroneous AS-i telegrams.



3.1.4 Double Master in IP20 with Power Supply A



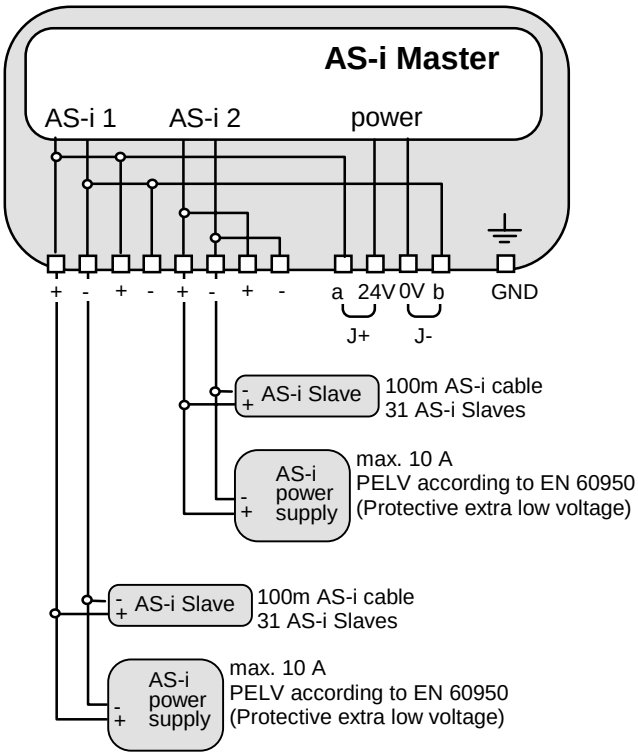
The terminals have the following functions:

+	"AS-i +", Actuator Sensor Interface 1 or 2, positive terminal These terminals are connected internally with point a2 of jumper "J+".
-	"AS-i -", Actuator Sensor Interface 1 or 2, negative terminal These terminals are connected internally with point b2 of jumper "J-".
24V	Master power supply, positive terminal (18 - 31.6 V DC)
0V	Master power supply, negative terminal
GND	Ground terminal
J+, J-	Jumpers for selecting the power supply of AS-i

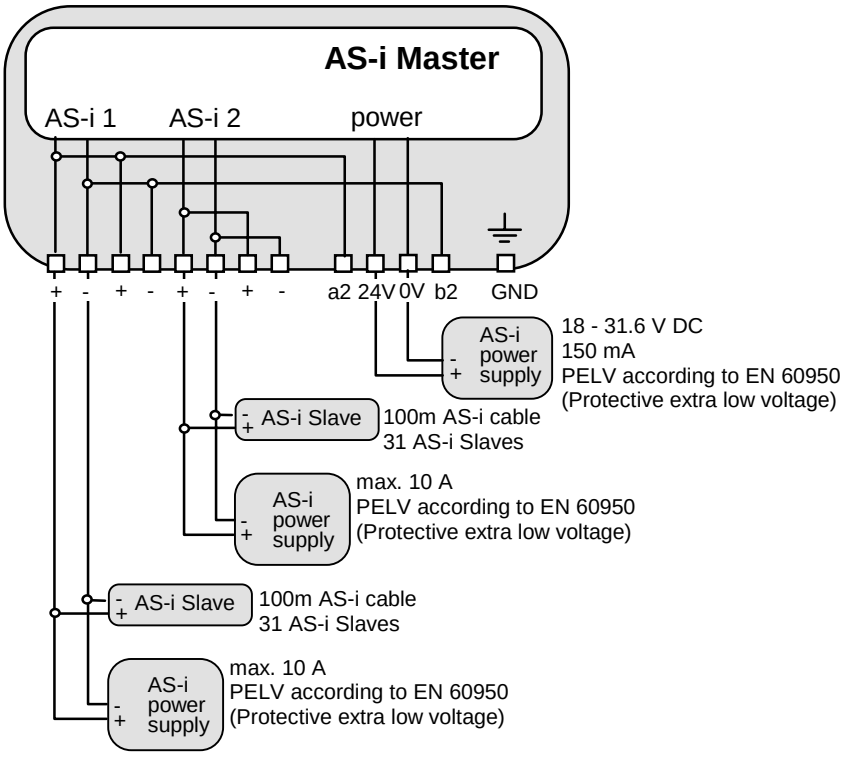
jumpers closed:
The AS-i Master is powered out of AS-i circuit 1.
Master power supply and AS-i network are then decoupled with coils.

jumpers open:
With the jumpers open (or missing), the AS-i Master must be powered by a separate 24 V DC power supply.

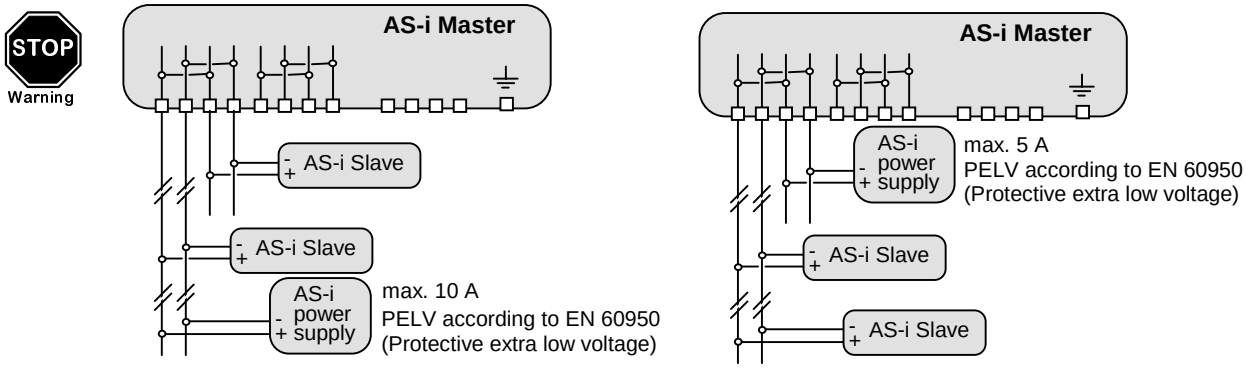
Power supply out of AS-i circuit 1



Operation with separate 24 V DC power supply

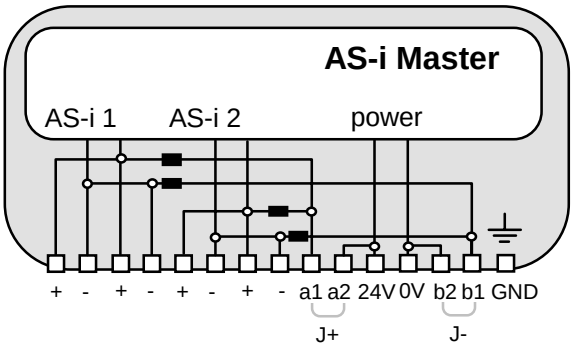


Connection variations for the AS-i circuits (here only displayed for one AS-i circuit



In the wiring schemes above the current through the AS-i Master must not exceed 5 A.

3.1.5 Double Master in IP20 with Power Supply N



The terminals have the following functions:

+	"AS-i +", Actuator Sensor Interface 1 or 2, positive terminal These terminals are connected internally with point a2 of jumper "J+".
-	"AS-i -", Actuator Sensor Interface 1 or 2, negative terminal These terminals are connected internally with point b2 of jumper "J-".
24V	Master power supply, positive terminal (18 - 31.6 V DC)
0V	Master power supply, negative terminal
GND	Ground terminal
J+, J-	Jumpers for selecting the power supply of AS-i

jumpers closed:

The AS-i circuits are powered by the master power supply.
Master power supply and AS-i network are then decoupled with coils.

jumpers open:

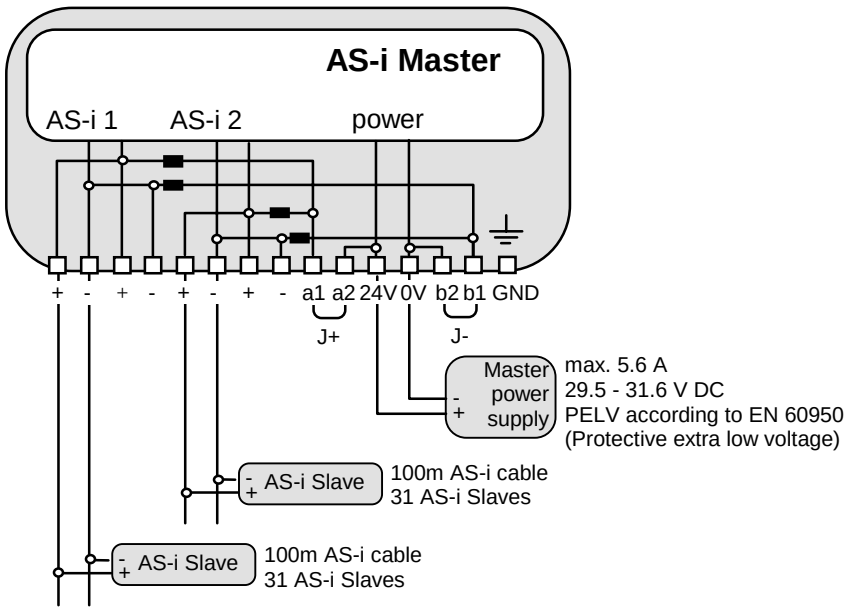
With the jumpers open (or missing), the AS-i Master must be powered by a separate AS-i power supply.
The AS-i power supply can be either connected to the jumper terminals or to the AS-i line in the field.

Operation without AS-i power supply

When the jumpers are closed, the AS-i networks are powered by the master's power supply. For test purposes, it is possible to use a conventional 24 V DC power supply to supply the AS-i network. You get optimum results with a 30 V DC voltage source.

Note

This variant requires only one power supply for 2 AS-i circuits!

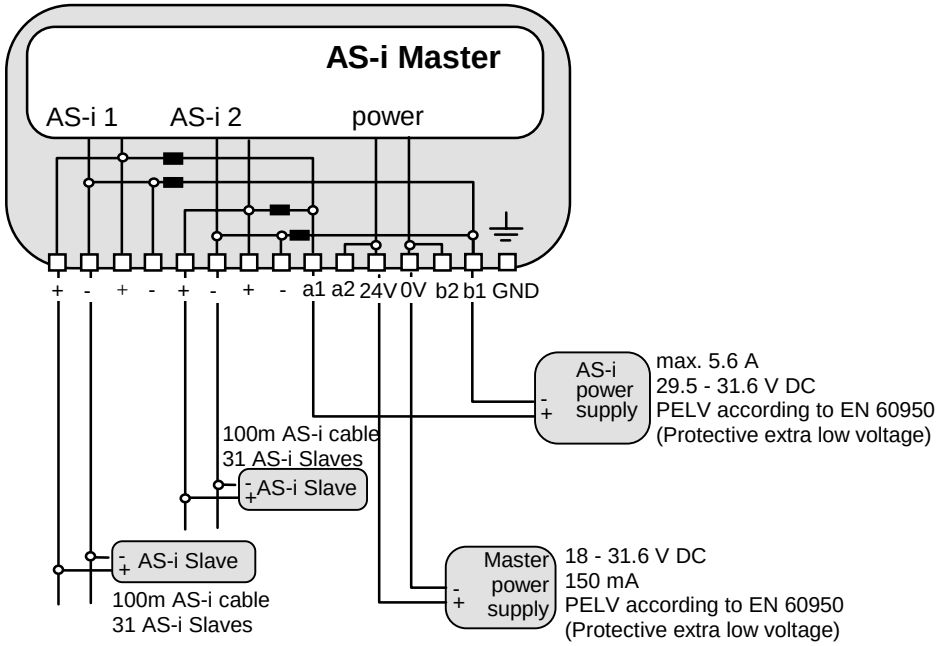


Attention

In the wiring schemes above the current through the AS-i Master must not exceed 2.8 A each AS-i circuit.

Operation with AS-i power supply

If the jumpers are open both AS-i circuits are power by one common AS-i power supply while the AS-i Master itself is powered by an additional 24 V DC power supply.

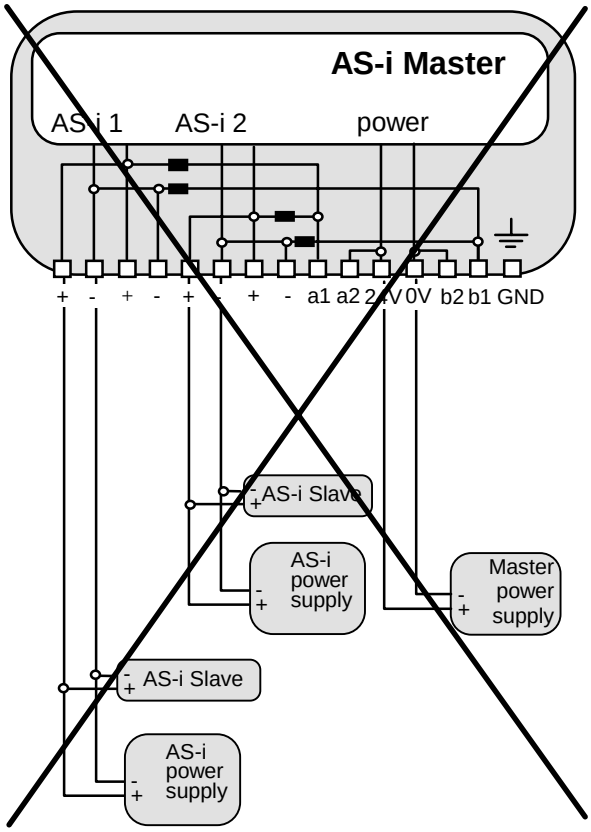


Attention

In the wiring schemes above the current through the AS-i Master must not exceed 2.8 A each AS-i circuit.

Warning

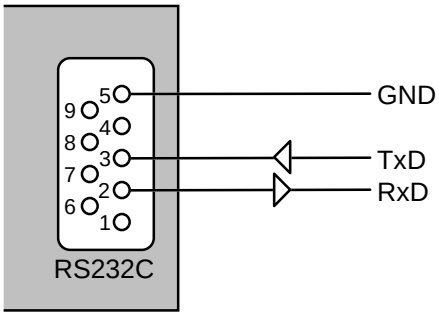
Do not try to provide the AS-i double master with power supply N with one AS-i power supply each AS-i circuit, because the additional decoupling coils would burden the AS-i line and cause erroneous AS-i telegrams.



3.2 The Serial Interfaces

The serial interface has been designed as a 9-pin sub-D type socket that is located on the right side of the front plate.

3.2.1 AS-i Master with RS232C

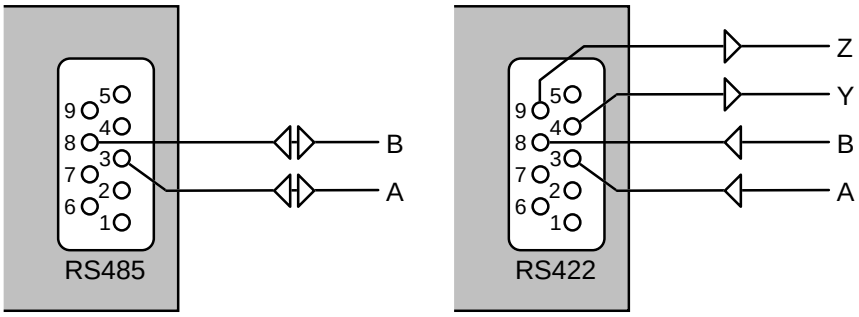


The AS-i Master with RS232C sends on pin 2 of the sub-D connector (“RxD” signal) and receives on pin 3 (“TxD” signal). Pin 5 of the sub-D connector carries the signal ground.

The collar of the connector and therefore the shield of the connector cable are connected galvanic with the ground terminal of the master.

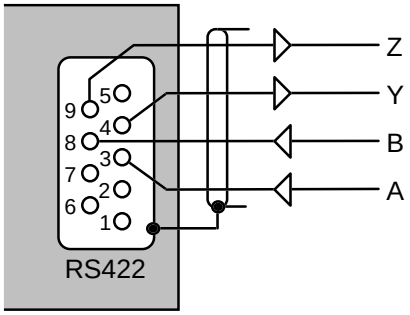
During the data transmission, the AS-i Master with RS232C acts as a DCE *(“Data Carrier Equipment”)*. When connected to a DTE (“*Data Terminal Equipment*”) device such as a PC, the connecting cable should be wired straight through without any crossed wires.

3.2.2 AS-i Master with RS485



AS-i Master with RS485 sends and receives on pins 3 and 8 of the sub-D socket. The RS485 signal “A” is assigned to pin 3 and signal “B” to pin 8.
To prevent compensation currents, the shield of the connecting cable is connected with the ground terminal of the master over a condensator. The shield should be grounded galvanic at an other place.

3.2.3 AS-i Master with RS422



AS-i Master with RS422 receives on pins 3 and 8 (“A” and “B”) and sends on pins 4 and 9 (“Y” and “Z”) on the sub-D socket.

Also here the shield of the conntecting cable is grounded only capazitiv.

3.3 Display and Operating Elements

On the front panel of the serial AS-i Master are seven light-emitting diodes, a two-digit display a two push buttons.

With the devices in IP65 the push buttons are situated inside of the housing to avoid liquids from entering. The top of the housing has to be screwed off to operate these push buttons.

3.3.1 LEDs of the Single Masters

In brackets you see the terms for the devices in IP65

power (PWR)
The master's power supply is sufficient.

ser active (BUSAKTIV)
LED on: Serial interface communication active via the serial interface
LED off: Serial interface communication not active via the serial interface
LED flashing: Serial interface communication active via the serial interface, the control program is active.

config err (CONF.ERR.)
Configuration error
At least one configured slave is missing, or at least one detected slave is not projected or for at least one projected and detected slave the actual configuration data does not match the nominal configuration data.

U ASI (U AS-i)
The AS-i circuit is sufficiently powered.

ASI active (AS-i ACTIVE)
Normal operation active.

prg enable	(PRG ENABLE) Automatic address programming enabled. Exactly one slave is missing in protected operating mode. The slave can be replaced by another slave of the same type with address zero. The master addresses the new slave to the faulty address and thus eliminates the configuration error.
prj mode	(PRJ MODE) The AS-i Master is in configuration mode.

3.3.2 LEDs of the Double Masters

AS-i 1/AS-i 2	Switching of the Displays and push buttons between the two AS-i circuits. If this LED lights up, all displays and button operations are related to AS-i circuit 2, otherwise AS-i circuit 1.
ser active	Serial interface communication is active via the serial interface By AS-i Control this LED signs the process of a controlling program additionally.
config err	Configuration error At least one configured slave is missing, or at least one detected slave is not projected or for at least one projected and detected slave the actual configuration data does not match the nominal configuration data.
power	The master's power supply is sufficient.
U AS-i	The AS-i circuit is sufficiently powered.
prg enable	Automatic address programming enabled. Exactly one slave is missing in protected operating mode. The slave can be replaced by another slave of the same type with address zero. The master addresses the new slave to the faulty address and thus eliminates the configuration error.
prj mode	The AS-i Master is in configuration mode.

4 Operating the Master

4.1 Peculiarities with Double Masters

Note

In **protected mode** the displays of the double masters are switched over from AS-i circuit 1 to AS-i circuit 2 in a measure of 2 seconds.

In **configuration mode** all detected AS-i slaves are displayed at first, before the display switches to the other AS-i circuit.

The operation of the push buttons is always related to the currently displayed AS-i circuit (LED AS-i 1/AS-i 2). After a push button was pressed the display stays with the respective AS-i circuit until the operation is finished or the operator has not interfered for 10 seconds.

4.2 Master Start-Up

After powering on, all segments of the number displays and all LEDs light up for approx. one second (self-test). Afterwards, the LEDs display the condition of their respective flags. The LCD displays the condition of the master according to the AS-i Master Specification:

Attention

40 Offline Phase
The AS-i Master initializes, there is no data communication on the AS-I. If the AS-I circuit is insufficiently powered (“U ASI” does not light up), the master stays in the offline phase.

41 Detection Phase
Start of the start-up phase, where the system looks for slaves located on the AS-I. The master stays in the recognition phase until it finds at least one slave.

42 Activation Phase
Condition at the end of the start-up operation where the parameters are transmitted to all connected and recognized slaves. This enables access to the AS-I slaves data connections.

43 Start of Normal Operation
In normal operation, the AS-I master can exchange data with all active slaves. It transmits management messages and looks for and activates newly connected slaves. During normal operation, the system maintains the maximum cycle time of 5 milliseconds.

4.3 Configuration Mode

Attention

In the configuration mode all recognized slaves are activated even when the desired and actual configurations do not match.

Pressing the “mode” button for at least five seconds switches the master to configuration mode. While in configuration mode, the yellow “prj mode” LED lights up.

The system then displays one after the other all detected slaves entered in the LDS („*List of Detected Slaves*“) at a speed of two per second. If the display is empty, no slaves were detached on the AS-i circuit.

In the configuration mode, all recognized slaves are activated except of slave zero. The AS-i Master is in normal operation. There is data exchange between the AS-i Master and all AS-i slaves detected by the master regardless of whether the detected AS-i slaves were projected before.

Attention

When delivered the AS-i/PROFIBUS Gateway is in configuration mode.

4.4 Protected Operating Mode

Note

In contrast with the configuration mode in the protected mode there is only data exchange between the AS-i Master and the projected AS-i slaves.

4.4.1 Switching to Protected Operating Mode

You leave the configuration mode by pressing the “mode” button.

Pressing the button shortly:

Exits the configuration mode without projecting the current AS-i configuration.

Pressing the button for more than five seconds:

Exits the configuration mode and projects the actual AS-i configuration. Simultaneously the actual AS-i configuration is stored as nominal configuration in the EEPROM.

Note

If the system detects an AS-i slave with address zero on the AS-i, it can not leave the configuration mode!

In the protected operating mode only AS-i slaves, that are projected in the LPS („*List of Projected Slaves*“), and whose actual configurations match the nominal configurations will be activated.

4.4.2 Configuration Errors in Protected Operating Mode

As long as there is no configuration error, the numeric display is turned off during the protected operating mode. Otherwise, the lowest address with a faulty assignment is displayed. A faulty assignment occurs when a slave has been recognized or configured (it has been entered on the LDS or on the LPS) but cannot be activated.

Pressing the “set” button displays the next higher faulty address.

4.5 Assigning an AS-i Address in Configuration Mode

AS-i can be put into operation in a very comfortable manner by using the Windows software AS-i Control Tools (see chapter 7.1)(addressing directly or with the AS-i address assistant).

For DOS users we have the DOS software AS-i SHELL.

Furthermore you can use a hand held addressing device.

If you don’t have neither a PC nor a hand held addressing device, address assigning of the AS-i slaves is also possible using the push buttons. How it works is described as follows.

4.5.1 Assigning a Slave Address

(assigning an available address to a slave with address zero)

In configuration mode, the addresses of all detected slaves are displayed one after the other. To display the next higher available operating address, press the “set” button shortly. Each time you press the “set” button, the next available address is displayed.

Chose the displayed address as your target address by pressing the button for more than five seconds. The address display blinks. The master is ready for programming; pressing the “set” button again addresses the connected slave with address zero to the target (blinking address).

Any errors will be displayed by their error codes according to chapter 4.3. Otherwise, the detected slaves are displayed again as described in chapter 4.3.

4.5.2 Erasing the Slave Address

(assigning address zero to a detected slave)

In configuration mode, the addresses of all recognized slaves are displayed one after the other. By pressing and releasing the “set” button, the master displays the next available address. If you press the button for more than five seconds while the address of a detected slave is displayed, this slave is will get the address zero and the display shows “00”.

When you release the button, the display continues to display the detected slaves.

4.6 Programming the Address in Case of Configuration Errors

4.6.1 Automatic Address Programming

Note

One of the AS-i's great advantages is the automatic address programming. When a slave fails, it can be replaced by one of the same type with address zero. The master will detect the replacement and automatically program the new slave with the address of the faulty one.

For automatic programming to work, some requirements must be fulfilled:

1. The AS-i Master must be in the protected operating mode.
2. The "auto_prog" release flag must be set.
3. Only one of the configured slaves may not be recognized.

When these requirements are fulfilled, the AS-i Master's "prg enable" LED lights up and a slave with address zero will be automatically assigned the operating address of the missing slave.

Attention

If the two slaves have different configuration data, i.e. are not of the same type as far as AS-i is concerned, the automatic address programming is not carried out.

4.6.2 Manual Address Programming

Note

If several slaves fail, they cannot be replaced automatically by the AS-i Master. You must set their addresses manually. If you don't want to set their addresses via the serial interface, you can set them with the keys and the display.

In the protected operating mode, wrong assignments are displayed as errors (see chapter 4.4). By pressing the "set" button, you can display all faulty assignments one after the other. By pressing the button for more than five seconds, you can select the currently displayed address as a potential target address, and the display starts to blink.

If the faulty slave was previously replaced by a slave with address zero, the new slave can now be programmed for the blinking address by pressing the "set" key again. As a requirement, the new slave's configuration data must match the configuration data for the blinking address.

After the address has been successfully set, the next faulty assignment is displayed or the display erased. Otherwise, the system displays an error code (see appendix).

4.7 Changing the Address of Bus-Capable Masters

You change the Address of the bus-capable AS-i Master with RS485- and AS-i Master with RS422 by pressing buttons „set" and „mode" for more than five seconds at the same time.

The actual bus address is shown then on the LCD display. Each time You press the "set" key, this address is incremented.

If the desired address is displayed, pressing "mode" for more than five seconds makes the master to accept the new address and and store it non-volatile in its EEPROM.

The address of the capable AS-i Master with RS485- and AS-i Master with RS422 may be changed via the serial interface also.

4.8 Error Messages

The system displays error codes for error messages that do not point to faulty assignments on the AS-i circuit. The code numbers are larger than 50 and are therefore outside the slave address range. These codes are described in the chapter 9.

5 Operation via the Serial Interface

5.1 Configuring the Interface

When transferring data via the AS-i Master's serial interface, the parameters must be set as follows:

Start bits	1
Data bits	8
Stop bits	1
Parity	none

The pin assignment for the sub-D connector is described in chapter 3.2.

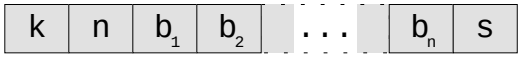
For the transmission speed, you can select 1200, 2400, 4800, 9600, 19200, 28800, 38400 or 57600 baud. If it has not received a valid host message since the last startup, the master automatically adapts to the host.

When selecting the baud rate, the master starts with the transmission speed that it used during the last communication with the host before it was turned off. As soon as a valid message is received, the baud rate remains fixed until the next startup.

5.2 Message Structure

The AS-i Master and the PC or PLC communicate with each other by exchanging messages. The host (PC or PLC in this case) functions as a master and the AS-i Master as a slave, i.e. the AS-i Master does not initiate any data communication but only responds to the host's messages.

The messages are structured as follows:



Command byte k:
: The first byte of each message is the command byte, that determines the AS-i function and therefore the message type.

User data length n:
: Indicates the number of user data bytes. Depending on the messages type, this number is between zero and 17.

User data bytes b_i:
: If no user data are to be transmitted with the message (usable data length n = 00_{hex}), this field is not used.

Checksum s:
: The lowest eight bits of the sum of all previously sent bytes are transmitted as the checksum. The checksum can also be calculated with the formula:

$$s = \left(k + n + \sum_{i=1}^{n} b_i \right) \bmod 256$$

The AS-i Master responds to a host message with a message of the same type but normally of different length, or it responds with an error message. (Command byte 75_{hex}, 1 byte usable data).

There can be some delay between host and slave messages since the master only responds after it has carried out the request it received with the message. The maximum processing times for the individual message types are shown in Appendix A. After the last character of the response message, the AS-i Master is ready to receive again.

Example:

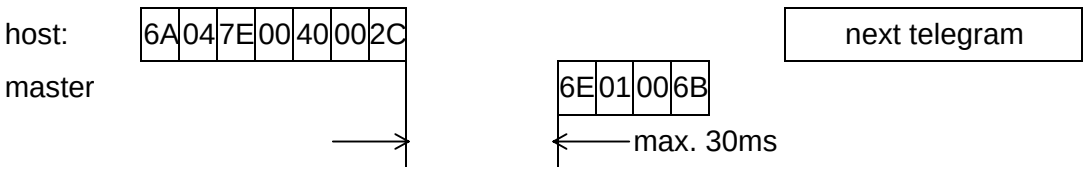
Addresses 1 through 6 and address 22 should be occupied in the list of projected slaves. The master is not in configuration mode, so it must not accept this request and answers with "*not o.k.*".

host message:

k $6A_{hex}$
n 04_{hex}
b_1 $01111110_{bin} = 7E_{hex}$
b_2 $00000000_{bin} = 00_{hex}$
b_3 $01000000_{bin} = 40_{hex}$
b_4 $00000000_{bin} = 00_{hex}$
s $6A + 04 + 7E + 00 + 40 + 00 = 12C_{hex} \Rightarrow s = 2C_{hex}$

master message:

k $6A_{hex}$
n 01_{hex}
b_1 "*not o.k.*" = 00_{hex}
s $6A + 01 + 00 = 6B_{hex}$



See chapter 8.1 for values of command byte, contents of data bytes for host- and master message and maximum processing times.

6 Advanced Diagnostics for AS-i Masters

The advanced AS-i diagnostics serves to locate occasionally occurring errors and to judge the quality of data transmission on AS-i without additional diagnostics tools.

The AS-i Control Tools, Bihl+Wiedemann's software for the comfortable commissioning of the AS-Interface and the programming of AS-i Control, will include the operation of the Advanced Diagnostics from version 3.0 on.

6.1 List of Corrupted AS-i Slaves (*LCS*)

To locate occasionally occurring short-time configuration errors the AS-i Masters with advanced diagnostics manage beside the list of projected slaves (*LPS*), the list of detected slaves (*LDS*) and the list of activated slaves (*LAS*) a forth list, the **list of corrupted slaves (*LCS*)**. This list contains entries of all AS-i slaves which were responsible for at least one configuration error since powering up the AS-i Master or reading the list. Short-time AS-i power failures are represented in the *LCS* at the position of AS-i slave with address 0.

With every read access the *LCS* will be deleted.

The last short-time configuration error can also be displayed on the AS-i Master: Pressing the "set" button of the AS-i Master shows the AS-i slave which was responsible for the last short-time configuration error. If a short-time AS-i power failure occured the display shows "39" after pressing the "set" button.

6.2 Error Counter: Counter of corrupted data telegrams

The AS-i Master with advanced diagnostics has an error counter for each AS-i slave, which is increased every time there is a corrupted AS-i telegram. This makes it possible to judge the quality of the AS-i network, even if only a few corrupted telegrams occurred and the AS-i slave did not cause any configuration errors.

The counter values can be read via the host interface and will be deleted with every read access. The counter value is limited to 254. 255 means counter overflow.

The error counter is included in the command Master | AS-i Diagnostics of AS-i Control Tools version 3.0.

6.3 Off-line Phase on Configuration Errors

The AS-i Masters with advanced diagnostics offer the possibility to put themselves into the Off-line Phase when a configuration error on the AS-Interface occurs. In this way the security of the application can be ensured. The reaction to a configuration error is very fast and the host can be relieved from this task. If there are any problems on the AS-i network, the AS-Interface can be switched to a secure state.

There are two different ways to parameterize the AS-i Master for this feature:

- Every configuration error during normal operation in protected mode releases the Off-line Phase.
- For each slave address can be chosen whether a configuration error on this address will release the Off-line Phase or not. This information is stored in the List of Off-line Slaves (*LOS*).
  The user himself can decide how the system reacts to a configuration error on the AS-interface. The AS-i Master can release the Off-line Phase in critical situations, i.e. only with certain slave addresses, while in less critical situations (if one of the other AS-i slaves have a configuration error) only the error message configuration error is sent to the host, but AS-i is still running.

The parameterization of Off-line Phase on Configuration Error is also supported by the AS-i Control Tools version 3.0.

7 Accessories for Putting AS-i into Operation and Test Tools

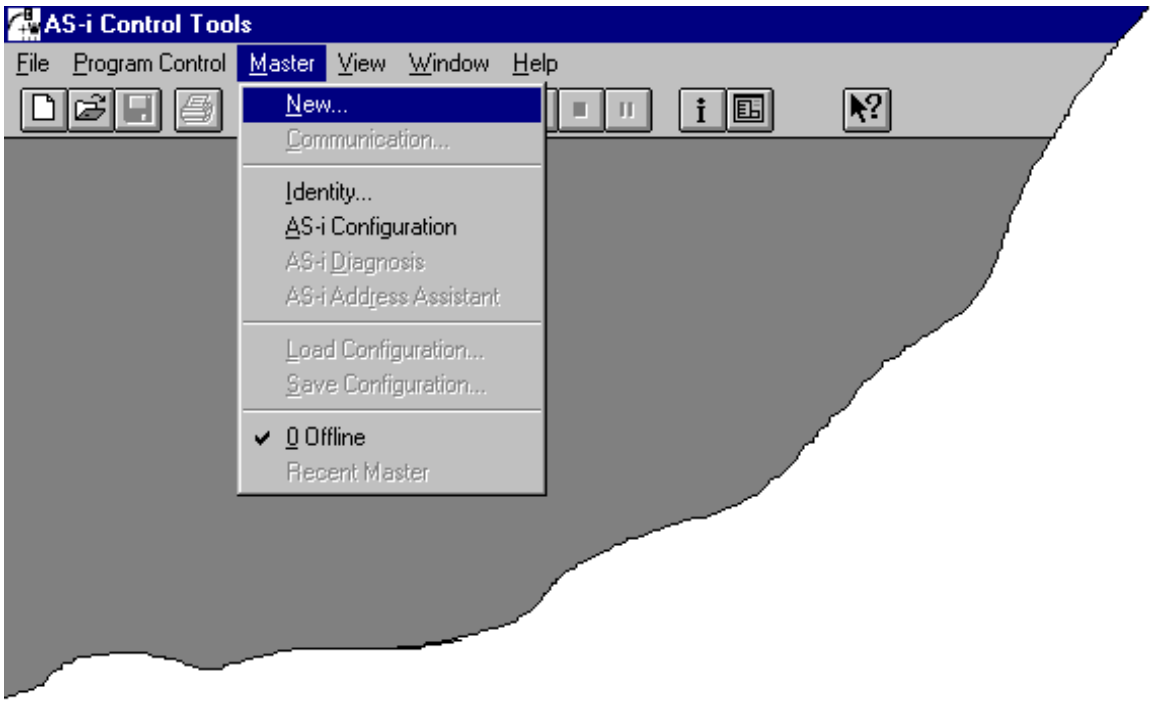
Putting AS-i into operation can be done very comfortably with the Windows software AS-i Control Tools. The DOS-software ASISHELL is also delivered with the AS-i Master. Both software packages communicate with the AS-i Master via a serial cable.

Furthermore the AS-i Master can be controlled with own programs with the serial telegrams decribed in chapter 8.

7.1 Windows Software AS-i Control Tools

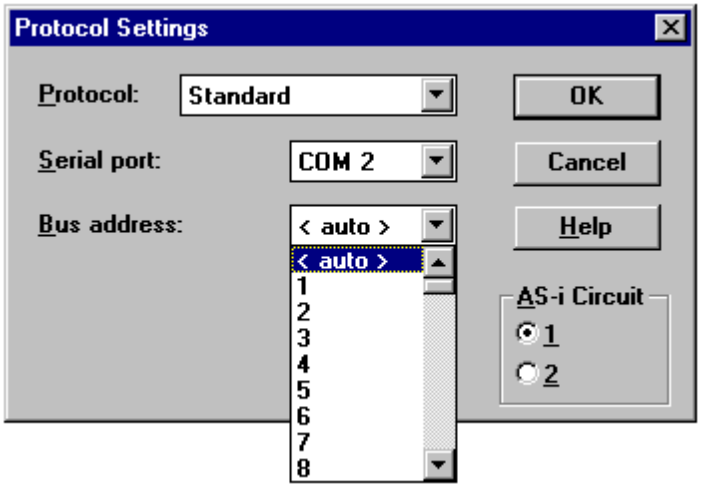
The Windows software AS-i Control Tools enables you to configure the AS-i circuit in a very comfortable manner.

1. For that purpose plug in the device to the D-SUB-connector and connect the device with a fully covered cable to the serial interface of your PC (RS232C, RS485, RS422).
2. Start the AS-i Control Tools.
3. Call the command Master | New.

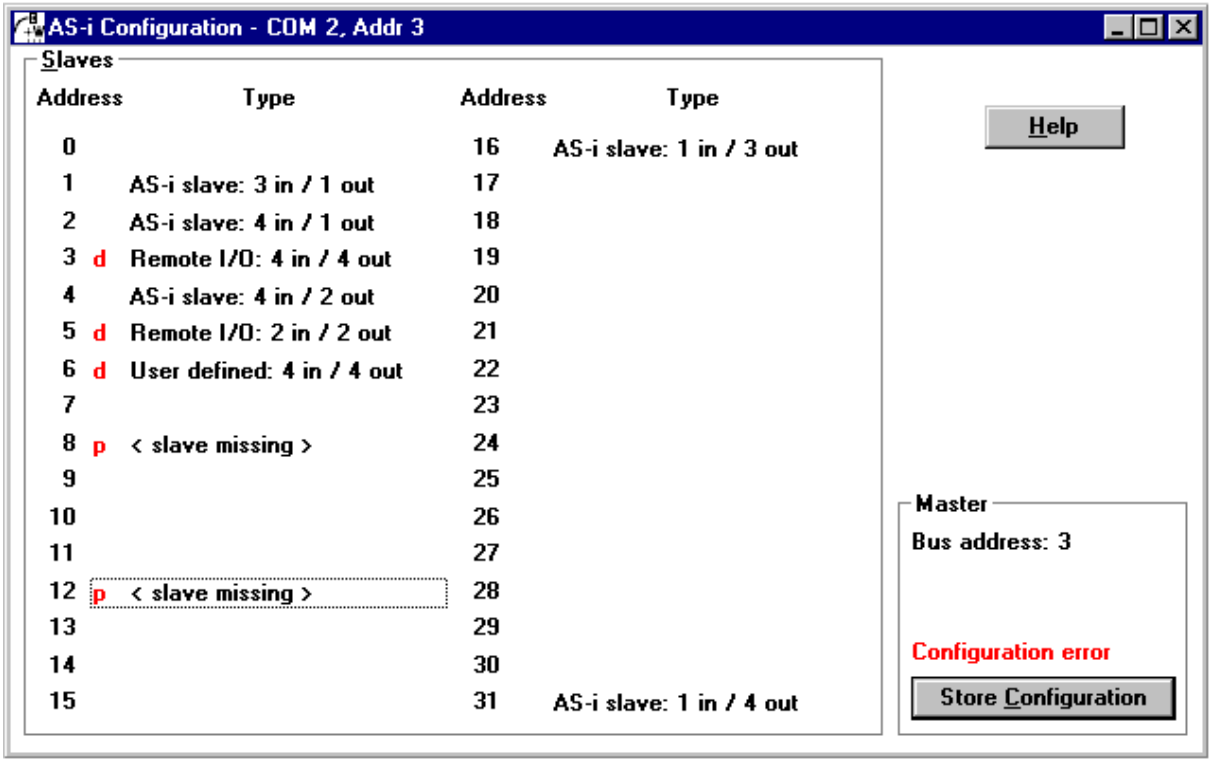


4. Choose the appropriate protocol (RS232C, RS485, RS422).

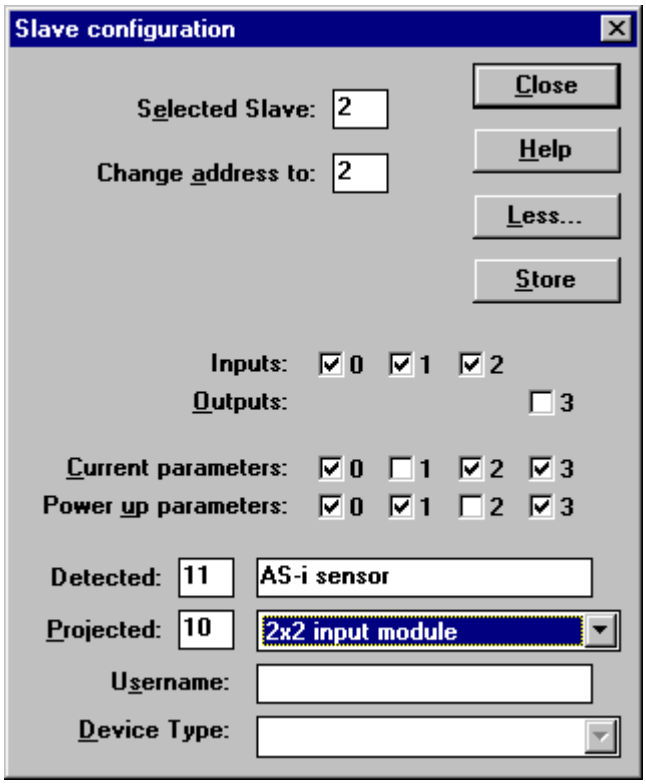
5. Do the appropriate settings (e.g. serial interface COM 2, station address <auto>, AS-i circuit 1)



6. Call the command Master | AS-i configuration.
   The AS-i configuration editor will be started. All detected and projected AS-i slaves are displayed in this window.


AS-i Configuration - COM 2, Addr 3
Slaves
Address Type Address Type
0 16 AS-i slave: 1 in / 3 out
1 AS-i slave: 3 in / 1 out 17
2 AS-i slave: 4 in / 1 out 18
3 d Remote I/O: 4 in / 4 out 19
4 AS-i slave: 4 in / 2 out 20
5 d Remote I/O: 2 in / 2 out 21
6 d User defined: 4 in / 4 out 22
7 23
8 p < slave missing > 24
9 25
10 26
11 27
12 p < slave missing > 28
13 29
14 30
15 31 AS-i slave: 1 in / 4 out
Help
Master
Bus address: 3
Configuration error
Store Configuration

7. Click on a slave entry to open the dialogbox slave configuration.



This dialog box is for changing a slave address, setting AS-i parameters or AS-i configuration data. Additionally you can test inputs and outputs.

A very easy approach to configure the AS-i circuit is connecting each AS-i slave to the line and setting the AS-i slave address one after the other. After that press the button “Store configuration“ to adopt the detected AS-i circuit to the AS-i Master as projected data.

Moreover you can use the **AS-i Address Assistant**. This tool changes automatically the address of an AS-i slave to the desired address after plugging the slave to the AS-i line. The desired AS-i configuration can be created off-line before and stored to a file. When you build up the plant you only have to plug the AS-i slaves to the AS-i line one after the other.

Further descriptions to all features of the software can be obtained from the integrated help.

7.2 DOS-Software

7.2.1 Frame Program ASISHELL

ASISHELL is an simple frame program to start the executables of the software package without knowledge of their command line parameters.

In the upper part of the menu items like the PC serial interface port or the name of the source code file for AS-i Control can be specified.

A second set of menu items is to process the controlling program by calling an text editor or ASI_CTRL.

ASI_VIS,. ASI_MON or ASI_ID can be executed by selecting one of the three menu items below.

A fourth block holds items to control the execution of ASISHELL:

- leaving ASISHELL,
- calling the DOS command line interpreter and
- showing the output of previous called programs.

The last menu item is to specify the command line to call the text editor. This makes it possible for every user to use the editor he is the most familiar with. Selecting the menu item „edit“ makes ASISHELL to pass this line to the operating system, after it replaced “$f” by the name of the source file and “$l” by the line number where ASI_CTRL found an error.

7.2.2 AS-i Monitor Program ASI_MON

The AS-i Monitor is a simple, screen-oriented MS-DOS program for monitoring, displaying (monitor mode) and editing (editor mode) of all process data and conditions of the serial AS-i Masters via a serial PC interface.

With the AS-i Monitor, the PC becomes an AS-i host system.

Starting the Program

In its default setting, the AS-i Monitor uses the PC’s COM1 serial port. If you want to use an other serial port, you must add the a appropriate parameter to the command line. For example, the command

```
ASI_MON 3
```

runs the program using COM3.

The Screen Design

In order to display the large amount of data orderly, two screen pages are used. On both pages, the current AS-i flag conditions are displayed at the top. The lower halves of both screens contain the slave-related AS-i data. Two footer lines display the function key assignments and status information. An additional screen page is available for logging any errors that might occur.

Status Messages

If necessary, status messages with the following meanings are displayed in the bottom left corner of the screen:

`*` Communication with the master is active.
(Same as the “ser active” LED on the master.)

`r` The master re-reads all AS-i data.
Normally, only the data that can be modified by changes in the AS-i circuit are updated. The master-internal data such as the configuration are read automatically only at the start and in longer intervals.

`p` One of the two AS-i functions “`Store_Actual_Parameters`” or “`Store_Actual_Configuration`” is currently active. This display was added since these functions can be carried also in the monitor mode with the function keys `F5` and `F6`.

`err` An error occurred during the last communication. You can display a detailed error description on the error screen.

Function Keys

The AS-i Monitor is controlled via function keys in both the monitor and the editor modes.

`F1` Help

`F3` Toggles between monitor and editor modes

`F5` Executes AS-i function “`Store_Actual_Parameters`”

`F6` Executes AS-i function “`Store_Actual_Configuration`”

`F9` Updates all AS-i data including the master-internal data from the master. This is necessary when you change the master, for example.

`F10` Exit AS-i Monitor

`Home, End PgUp, PgDn`
Change the screen page; the error display screen is the last page

Editor Mode

In the editor mode, you cannot only display all process data and conditions, but also change them. Just move the cursor to the information you want to change and overwrite it.

You can move the cursor only to the fields that can actually be changed. Any valid entries are executed immediately.

Except for the two functions “`Store_Actual_Parameters`” and “`Store_Actual_Configuration`” that are executed with function keys, all control level functions of the current AS-i Master Specification can be modified.

Changing Hexadecimal Information

Since they have a value range between zero and 15, input and output data, slave parameters and configuration data are displayed in hexadecimal notation.

In order to change these values, move the cursor on editor mode to the respective position and enter the new value as a single-digit hexadecimal value. Any entries that cannot be interpreted as hexadecimal values are ignored.

Changing Binary Information

The three slave lists (*LDS*, *LAS* and *LPS*) and the execution level flags (in the AS-i Master) are displayed in binary notation. A set bit is represented by an "X", and the erased bits are represented as a period.

After you move the cursor to the proper position, press the spacebar to toggle the values.

Changing the Slave Addresses

A slave address can be overwritten by entering a two-digit decimal number. Invalid keystrokes are ignored, but both digits must be entered one after the other.

7.2.3 Tokenizer/Loader ASI_CTRL

The ASI_CTRL program serves to read the controller program from a source text file and convert it to a code that can be read by AS-i Control. This process is called "*tokenizing*".

The generated code is not written to a new file, but can - if desired - be stored directly in the AS-i Control's EEPROM. Simultaneously, the AS-i configuration contained in the Setup section of the source text file is entered into the AS-i Control configuration data ("downloading").

The loaded controller program can optionally be activated in the AS-i Control.

During the downloading, the following AS-i host functions are executed:

- Slaves listed in the *Setup* sections are entered into the list of projected slaves (*LPS*).
- The parameters in the *Setup* sections (default F_{hex}) are sent to all slaves in the *LPS*.
- These parameters are entered into the AS-i Control permanent parameters list.
- I/O codes and configuration IDs for these slaves are entered into the AS-i Control permanent configuration list.

The controller program is stored in the AS-i Control's EEPROM. Before such a program can be started, it must be reset (read from the EEPROM).

In addition to the source text file name, the following parameters can be added when calling the ASI_CTRL program:

`/p`<*number*> "select serial port"

Selects the serial PC interface port that is to be used for the communication with AS-i Control. <*number*> can be 1 through 4 (for `COM1` through `COM4`).
Example:
"`asi_ctrl demo.asi /p1`" converts the source text file `demo.asi` and uses `COM1` to communicate with AS-i Control.

`/b`<*number*> "select baud rate"

Selects the serial transmission rate in bits per second. The valid baud rates are 1200, 2400, 4800, 9600, 19200, 38400 and 57600 bits per second.
Example:
"`asi_ctrl demo.asi /b19200`" converts the source text file `demo.asi` and transmits to AS-i Control at a speed of 19200 baud.
This setting can be stored as a default value with `/w`.

`/m`<*number*> "select master bus-address "

The AS-i Masters with RS485 and RS422 interface are bus-capable. `/m` selects the bus address of the master to work with.
If you use the non-bus-capable AS-i Master with RS232C-interface, you have to specify "`/m-`".

`/d+` or `/d–` "download"

This switch determines whether ASI_CTRL should only check the source text file for correct syntax (`/d+`) or also store it in the AS-i Control's EEPROM (`/d–`).

`/u+` or `/u–` "upload"

This switch determines whether ASI_CTRL should read the controller program in ASI_CTRL and write it to the source text file (`/u+`).

`/s+`, `/s–` or `/s0` "start control program"

Determines whether the controller program loaded with /d+ should be activated (`/s+`), stopped (`/s–`), or whether the start flag condition should remain unchanged (`/s0`).
When a controller program is deactivated with `/s–`, AS-i Control acts like a normal serial AS-i Master from Bihl+Wiedemann even during normal operation in the protected operating mode.

`/v+` or `/v–` "produce verbose output"

`/v+` causes a very detailed screen output that displays the individual steps during the token generation and the download to AS-i Control.
Independent of `/v+` and `/v–`, the text file `ASI_CTRL.LOG` is generated each time ASI_CTRL is called. This file contains these detailed messages.

`/o+` or `/o-` "overwrite source file"

(makes sense only in combination with `/u+`).
When the name of an existing source text file is indicated during the upload, it is normally overwritten after an additional confirmation by the user.
This process can be suppressed with `/o+`.

`/h` or `/?` "display help"

Displays help text with the various settings and the default settings stored with `/w`.

`/w` "write ini-file"
Stores the settings made with `/p`, `/b`, `/d`, `/u`, `/s`, `/o`, and `/v` as default values in the file `asi_ctrl.ini`.

7.2.4 Visualizer ASI_VIS

The ASI_VIS program displays the AS-i circuit's input and output data and the AS-i Control's user memory while a controller program is running.

The command parameters have the following functions:

`/p`<*number*> "select serial port"
Selects the serial PC interface port that is to be used for the communication with AS-i Control.
<*number*> can be 1 through 4 (for `COM1` through `COM4`).
Example:
"`asi_vis /p1`" uses `COM1`.

`/b`<*number*> "select baud rate"
Selects the serial transmission rate in bits per second. The valid baud rates are 1200, 2400, 4800, 9600, 19200, 38400 and 57600 bits per second.
Example:
"`asi_vis /b19200`" transmits to AS-i Control at a speed of 19200 baud.

`/m`<*number*> "select master bus-address "
The AS-i Masters with RS485 and RS422 interface are bus-capable. `/m` selects the bus address of the master to work with.
If you use the non-bus-capable AS-i Master with RS232C-interface, you have to specify "`/m-`".

`/h` *or* `/?` "display help"
Displays help text with the various settings and the default settings stored with `/w`.

8 Including the AS-i Masters in Own Programs

The AS-i Master can be directly responded by own programs with the help of the serial telegrams. There are three methods to do this:

1. Direct responding to the AS-i Master from own programs with the help of the serial telegrams, described in the following chapter 8.2.
2. Using of the delivered PC drivers under MS-DOS (see chapter 8.2).
3. If the environment is Windows: Using DLLs or the Windows NT driver of Bihl+Wiedemann.

However we recommend to use the first method.

8.1 Telegrams of the Serial Communication

8.1.1 Message Structure

The messages have the following structure:

k	n	b_1	b_2		...		b_n	s

Command byte k:

Message ID character.

User data length n:

Number of user data bytes (zero to 17).

User data bytes b_i:

If user data length n = 00_{hex}, this field is not used

Checksum s:

The lowest eight bits of the sum of all previously sent bytes are transmitted as the checksum. The checksum can also be calculated with the formula:

$$s = \left(k + n + \sum_{i=1}^{n} b_i \right) \bmod 256$$

The AS-i Master responds to a host message with a message of the same type but normally of different length, or it responds with an error message (command byte 75_{hex}, 1 byte usable data).

Example:

For a change of the operating address from 7 to 26, the messages would look like this:

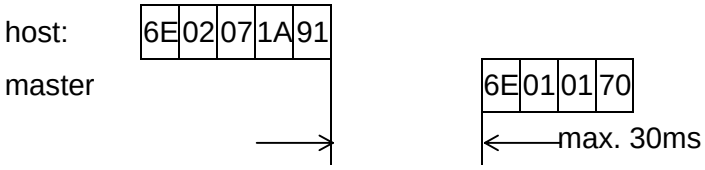
Host message:

command byte k:	$6E_{hex}$
user data length n:	02_{hex}
user data byte b_1 :	*old slave address* = 07_{hex}
user data byte b_2:	*new slave address* = $1A_{hex}$
checksum s:	$6E + 02 + 07 + 1A = 91_{hex}$

Master message (master responds with “O.K.”)

command byte k:	$6E_{hex}$
user data length n:	01_{hex}
user data byte b_1:	*status* = O.K. = 01_{hex}
checksum s:	$6E + 01 + 01 + = 70_{hex}$

maximum reaction time of the master: 30 ms



8.1.2 Message Descriptions

In tables of the following pages are listed for each communication message the command byte k, the content of the data byte b_i for host and master massage and the maximum reaction time t_{max} of the master.

The recommendable communication messages are printed bold.

Commands according to the AS-i Master Specification				
message	k	b_i (host message data)	b_i (master message data)	t_{max}
read input data	71_{hex}	-	b_1...b_{16}: input data	10ms
write output data	70_{hex}	b_1...b_{16}: output data	b_1: status	10ms
write configured parameters	61_{hex}	b_1: slave address b_2: parameters	b_1: status	30ms
read configured parameters	62_{hex}	b_1: slave address	b_1: parameters	20ms
write actual parameters	63_{hex}	b_1: slave address b_2: parameters	b_1: counter-read parameters (inverted in case of error)	20ms
read actual parameters	64_{hex}	b_1: slave address	b_1: parameters	20ms
store actual parameters	65_{hex}	-	b_1: status	200ms
write configuration data	66_{hex}	b_1: slave address b_2: configuration data	b_1: status	30ms
read configuration data	67_{hex}	b_1: slave address	b_1: configuration data	10ms
store actual configuration	68_{hex}	-	b_1: status	200ms
read actual configuration	69_{hex}	b_1: slave address	b_1: configuration data	10ms
write list of projected slaves	$6A_{hex}$	b_1 ... b_4: LPS	b_1: status	30ms
read list of projected slaves	$6B_{hex}$	-	b_1 ... b_4: LPS	10ms
read list of activated slaves	$6C_{hex}$	-	b_1 ... b_4: LAS	10ms
read list of detected slaves	$6D_{hex}$	-	b_1 ... b_4: LDS	10ms
read execution control flags	72_{hex}	-	b_1: execution control flags	10ms
set operating mode	73_{hex}	b_1 = 0: protected operating mode b_1 = 1: configuration mode	b_1: status	100ms
write host interface flags	74_{hex}	b_1: host interface flag	b_1: status	30ms
change slave´s operating address	$6E_{hex}$	b_1: old slave address b_2: new slave address	b_1: status	30ms
AS-i command call	$6F_{hex}$	b_1: slave address b_2: information part of the master request	b_1: response from slave b_2: status	30ms

Additional Commands (not in AS-i Master Spezification)				
message	k	b_i (host message data)	b_i (master message data)	t_{max}
exchange all input and output data	**76_{hex}**	**$b_1...b_{16}$: output data**	**b_1: execution control flags $b_2...b_{17}$: input data**	**10ms**
write selected output data	77_{hex}	b_1: 1st slaveadresse b_2: amount of slaves $b_3...b_{18}$: output data	b_1: status	10ms
read selected output data [2]	78_{hex}	b_1: 1st slave address b_2: amount of slaves	b_1: execution control flags $b_2...b_{17}$: input data	10ms
read output data	81_{hex}	-	$b_1...b_{16}$: output data	10ms
write parameter field	$8C_{hex}$	b_1: slave address b_2: actual parameters	b_1: status	10ms
read configured data of all AS-i circuits	$8D_{hex}$	b_1: number of the AS-i circuit b_2: slave address	b_1: status b_2: configured parameter b_3: configured data	10ms
configure all AS-i circuits	$8E_{hex}$ $8D_{hex}$	Request 1 (start): $b_1...b_2$: FF_{hex} $b_3...b_4$: 00_{hex} Request 2 (data): b_1: number of the AS-i circuit b_2: slave address b_3: parameter of the slave b_4: configured data of the slave Request 3 (commit): $b_1...b_2$: FF_{hex} $b_3...b_4$: 01_{hex} -- b_1: status		300 ms --
read master version	$7D_{hex}$	b_1: ≡ 0: versions number (8 Bytes) b_1: ≡ 1: master name part 1 (17 Bytes) b_1: ≡ 2: master name part 2 (17 Bytes) b_1: ≡ 3: master version (17 Bytes) b_1: ≡ 4: installied software and host interface flags (17 Bytes)	b_1: version information (8 oder 17 bytes)	10ms
activate/deactivate watchdog[1] for serial communication	$7B_{hex}$	b_1= 0: deaktiviert watchdog $b_1 \neq 0$: watchdog-timeout * 10ms	b_1: status	10ms
read watchdog status for serial communication	$7A_{hex}$	-	b_1= 0: watchdog not aktive $b_1 \neq 0$: max. watchdogtime * 10ms	10ms

[1] If the Watchdog has been activated, AS-i will go in Offline-Phase. By sending this message again AS-i leaves the Offline Phase.

lock/unlock front panel operation	$7C_{hex}$	b_1= 0: front panel operation enabled b_1= 1: front panel operation disabled	b_1: status	10ms
error message	75_{hex}	only sent by the master!	b_1: error code	-

Additional Commands for RS232C- Masters				
message	k	b_i (host message data)	b_i (master message data)	t_{max}
activate master	$7E_{hex}$	b_1, b_2: address of the master to be activated	b_1: status	20ms

Additional Commands for RS485- und RS422- Masters				
message	k	b_i (host message data)	b_i (master message data)	t_{max}
activate master	$7E_{hex}$	b_1, b_2: address of the master to be activated	b_1: status	20ms
change master address	82_{hex}	b_1: old master address b_2: new master address	b_1: status	30ms

Additional Commands for AS-i Double - Masters				
message	k	b_i (host message data)	b_i (master message data)	t_{max}
Exchange all process data[2] (write *ODI*, read *ec*-flags *IDI*)	**$8B_{hex}$**	**b_1...b_{16}: output data AS-i circuit 1 b_{17}...b_{32}: output data AS-i circuit 2**	**b_1: *ec*-flags AS-i circuit 1 b_2...b_{17}: input data AS-i circuit 1 b_{18}: *ec*-flags AS-i circuit 2 b_{19}...b_{34}: input data AS-i circuit 2**	**10ms**

[2] This message makes no difference, which of the AS-i circuits has been activated.

Additional Commands for AS-i Control				
message	k	b_i (host message data)	b_i (master message data)	t_{max}
write 16 controller program bytes (download)	$7F_{hex}$	b_1, b_2: start address b_2...b_{18}: 16 bytes of the controller program	b_1: satus	200ms
read 16 controller program bytes (upload)	83_{hex}	b_1, b_2: start adress	b_1...b_{16}: 16 bytes of the controller program	10ms
read AS-i Control status	83_{hex}	b_1, b_2: $FFFF_{hex}$	b_1: AS-i Control flags b_2: 00_{hex} b_3, b_4: current cycle time b_5, b_6: maximum cycle time	10ms
start/stop controller program	80_{hex}	b_1: start/stop code	b_1: status	20ms
reset controller program	80_{hex}			3000ms
read user memory (flags)	84_{hex}	b_1: start address b_2: amount of bytes to be transmitted (max. 16)	b_1...: user memory	10ms
write user memory (flags)	85_{hex}	b_1: start address b_2: amount of bytes to be transmitted (max. 16) b_3...: user memory	b_1: status	10ms

Commands for Advanced Control				
message	k	b_i (host message data)	b_i (master message data)	t_{max}
advanced diagnostics	88_{hex}	b_1: selection	n=0: b_1-b_{15}: slave 1 - 31 n=1: b_1-b_{15}: slave 0 - 15 n=2: b_1-b_{15}: slave 16 - 31	10ms
write list of off-line slaves *LOS* (off-line by configuration error)	89_{hex}	b_1 -b_4: slaves 0 - 31	b_1: error status	10ms
read list of off-line slaves *LOS* (off-line by configuration error)	$8A_{hex}$	-	b_1 -b_4: slaves 0 - 31	10ms

Commands for Backward Compatibility with Older Master Versions				
message	k	b_i (host message data)	b_i (master message data)	t_{max}
enable/disable automatic programming	79_{hex}	$b_1 \equiv 0$: disable $b_2 \equiv 1$: enable	b_1: status	30ms

8.1.3 Representation of Information in the User Data Bytes

Input and Output Data

For each slave, a four-digit binary number can be entered as input and output data. Input and output data can therefore range between 0 and 15.

For serial transmission, the data for two slaves are combined in a single byte. With message „q“ (read input data), the master therefore sends 32/2 = 16 bytes of user data.

The entries for low slave addresses are transmitted first. Byte 0, bits 0 through 3 (lower nibble) thus contains the input data of the slave with operating address zero; the upper nibble of the user data byte 15 contains the data of slave 31.

Slave Lists

The AS-i Slave lists LPS, LDS, LAS, LCS and LOS are built up as follows:

byte	0								1							
bit	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
slave	-	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

byte	2								3							
bit	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
slave	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

Meaning of the Lists:

LPS List of Projected Slaves

LDS List of Detected Slaves

LAS List of Activated Slaves

LCS List of Corrupted Slaves
List of those slaves, that have caused a short-time configuration error.

LOS List of Off-line Slaves
List of those slaves, with that in case of configuration error the AS-i Master shall switch to the Off-line phase.

Execution Control Flags

The execution control flags are transmitted in the diagnosis telegram, if the gateway is operated in the professional mode.

When set, the individual bits have the following meaning:

Bit 0:	*Config_OK*	no configuration error
Bit 1:	*LDS.0*	slave with address 0 present
Bit 2:	*Auto_Address_Assign*	automatic programming permitted
Bit 3:	*Auto_Address_Available*	automatic programming available
Bit 4:	*Configuration_Active*	configuration mode active
Bit 5:	*Normal_Operation_Active*	normal operation active
Bit 6:	*APF*	AS-i power failure
Bit 7:	*Offline_Ready*	off-line mode active

Host Interface Flags

The set of the host interface flags has follwing effects:

Bit 0:	*Data_Exchange_Active*	if bit 0 is set, the data communication between master and slaves is disabled (*the flag is transmitted inverted*)
Bit 1:	*Off-line*	if bit 1 is set, the master is put in offline mode
Bit 2:	*Auto_Address_Enable*	if bit 2 is set, automatic programming is disabled (*the flag is transmitted inverted*)

Installed Software/Host Interface Flags (message 7D$_{hex}$)

If message 7D hex (“read master version”) is sent with a “4” in the host message’s data byte, the AS-i Master responds with a 17 bytes long character string (16 letters, null-terminated).

The letters have the following explanations:

Byte 0 (C/c, D/d)
The responding AS-i Master is an AS-i Control.
The capital ‘C’ means that a controller program is currently being executed.
A lower-case ‘c’ means that either the start flag has not been set or that the AS-i Master’s status does not permit the execution.
If D/d instead of C/c is displayed, it is the newer software version of AS-i Control II.

Byte 1 (B/b)
The responding master has a bus-capable RS485 or RS422 interface. The 7E$_{hex}$ (activate master) and 82$_{hex}$ (change master address) messages can be processed.

Byte 2 (F/f)
The responding AS-i Master is featured with an AS-i error counter.

Byte 3 (E/e)
The responding AS-i Master is featured with an emc test mode.

Byte 4 (D/d)
The responding AS-i Master is featured with advanced diagnostics.

Byte 5 (C/c)
The responding AS-i Master is featured the function off-line by configuratiion error.

Byte 6 (./2)
The responding AS-i Master manages one (´.´) or two (´2´) AS-i circuits.

Byte 7 not used

Byte 8 (D/d)
The “data_exchange_active” host interface flag is set/erased.

Byte 9 (O/o)
The “off-line” host interface flag is set/erased.

Byte 10 (A/a)
The “auto_address_enable” host interface flag is set/erased.

Byte 11 - 13 not used

Byte 14 (W/w)
The serial watchdog was activated/deactivated.

Byte 15 (T/t)
The operation of the AS-i Master via the front panel buttons is enabled/disabled.

AS-i Control Flags, Start/Stop Code

Bit 0:	*start_flag*	if bit 0 is set, the controller program is executed as soon as the AS-i Master's status permits. (This flag is stored non-volatile.)
Bit 1:	*reset_bit*	the controller program is read from the EEPROM prior to the start. In addition, the user memory (flag bytes) is erased. (Necessary after each download), not returned as AS-i Control flag.)
Bit 2:	*ignore_config_errors*	if bit 2 is erased, the controller program is stopped as soon as an AS-i configuration error occurs. (This flag is stored non-volatile.)
Bit 3:	*auto_start*	if bit 3 is set, AS-i Control waits for a push on the "set" button before it restarts the controller programm. (This flag is stored non-volatile.)
Bit 4:	*counter_map*	if bit 4 is set, the counter registers of the 15 counters can be accessed by `M 96.0` to `M 125.7`. (This flag is stored non-volatile.)

Status Byte

The master returns the status byte, if there would otherwise be no user data. Normally, it takes on only one of the two following values:

`00` error while executing a host request

`01` no error while executing a host request

When changing a slave address, detailed error messages are returned:

`02` slave whose address should be changed does not exist

`03` slave with operating address 0 present

`04` address for which the slave should be programmed is already occupied

`05` slave could not be programmed to address 0

`06` slave could not be set for new operating address

`07` new operating address could not be stored in slave's EEPROM

Error Codes in the error message

Bit 0: checksum error

Bit 1: time-out

Bit 2: unknown command

Bit 3: illogical message length

Bit 4: illogical number of user data bytes

Bit 5: watchdog timer expired

Bit 6: command execution error

By setting several bits, more than one error can be reported.

8.2 PC Drivers

AS-i Master and PC communicate together through exchange of communication messages. This communication can token on with the delivered driver routines.

The including of the driver in own programs demonstrate the simply example routines „demo.c“ and „ demo2.c“ on the delivered disk.

8.2.1 New Data Types

The only new introduced data type is `ASI_DATA`. `ASI_DATA` is a field with elements of the unsigned char that is used to accept the user data. You can access the individual elements of this data field with the customary C subscripts. „data[3]“, for example, is the third element in the `ASI_DATA` data field.

You can also pass your own data structures to the communication routines. Using `ASI_DATA`, however, ensures a sufficient data field size.

8.2.2 Global Static Variables

Some information, that is normally not needed by the main program, is stored in ststic variables. In most cases, these are status informations for the most recent transmissions.

All of these variables execept ASI_RETRY are set by the driver and should only be read by the main program.

ASI_error	last error status (the last teturn value different of zero of ASI_com ())
ASI_error_tel	Content of the last error massage
ASI_message	kind (AS-i function) of the last sent message of the PC
ASI_message_len	itslength
ASI_message_chk	its checksum
ASI_answer	kind of the last response of the master
ASI_Answer_len	its length
ASI_Answer_chk	its checksum

8.2.3 Functions

int ASI_init (int port, int baud);

This function must be called prior to the first communication since it initializes the driver and the serial interface. For the port parameter, the constants “COM1” through “COM4” are defined in the asi_ser.h header file. The valid baud rates are 1200, 2400, 4800, 9600, 19200, 28800, 38400, and 57600 bits per second. The master automatically recognizes the baud rate set on the PC.

The serial interface is set for 8 data bits, no parity, and one start and stop bit each.

After the first successful communication with the master, the system tries to run the serial interface interrupt-driven which increases the reliability with low computing requirements and high baud rates (for example at 9600 baud in the Windows DOS emulator). If this is not possible, the interface module is accessed by the other driver routines via polling.

Return value:

- 00_{hex} ok
- 03_{hex} AS-i Master does not respond
- 04_{hex} Cannot understand the master’s response (possibly caused by differing baud rates)
- 81_{hex} COM port number invalid
- 82_{hex} COM port not found
- 83_{hex} Insufficient DOS memory

84_{hex} Serial interface module cannot run at this baud rate

`int ASI_com (char command, ASI_DATA message, ASI_DATA answer);`

This is the general function for the communication with the master. A message of the "`command`" type is sent to the master. The user data are transferred in `message`, and the user data of the response are stored in `answer`.

The return value's low bytes (hexadecimal) have the following meanings:

00_{hex} error-free communication

01_{hex} `ASI init()` had not yet been called

02_{hex} invalid AS-i command in `command`

03_{hex} Master responded with a wrong message

04_{hex} Master's response message had invalid length

05_{hex} the response message had a wrong checksum

06_{hex} the low-level communication function reports an error

10_{hex} error message received

In case of an interface error, the error code of the low-level communication function is returned in the higher-value byte of the return value. If more than one error occurred, the sum of the individual error codes is returned.

01_{hex} time-out

02_{hex} overrun error

04_{hex} parity error

08_{hex} framing error

10_{hex} error occurred during the response's first character

FF_{hex} internal buffer overrun

`int ASI_is_command (char command);`

This function is used to check for a valid message type.

Return value:

1_{hex} command is a valid message type

0_{hex} command is an invalid message type

`int ASI_get_message_len (char command, ASI_DATA message);`
`int ASI_get_answer_len (char command, ASI_DATA message);`

With these functions, you can determine the number of user data bytes in a send or response message.

For message types that permit a variable message length, the actual length is determined by the data actually transmitted in `message` or `answer`.

If `message` or `answer` are unknown, `NULL` can be transmitted. Both functions then return the maximum number of data bytes.

9 Appendix A Seven-Segment Display

In the basic configuration mode, the display shown one after the other the addresses of all detected slaves at a rate of two per second. A blank display means that the LDS is empty, i.e. no slaves were recognized.

In the basic protected operating mode, the display is either blank or displays the address of a faulty assignment (see chapter 4.4.2).

During manual address programming, the slave address display has a different meaning, of course (see chapter 4.5.1 and 4.5.2).

All displayed numbers that are larger than 31 and can therefore not be interpreted as slave addresses are status or error messages of the master. They have the following meanings:

40	AS-i Master is in offline mode
41	AS-i Master is in detection mode
42	AS-i Master is in activation mode
43	AS-i Master starts normal operating mode
66	baud rate is being switched.
70	the AS-i Master's EEPROM cannot be written to.
72	the interrupt bit of PIC processor remains set
73	the PIC processor does not respond
80	slave with address zero detected. The system attempted to exit the configuration mode with a slave zero.
81	General error while changing a slave address
82	Front panel operation blocked. Until the next power-up the only way of accessing the master is via the serial interface.
88	Display test while starting up the AS-i Master
90	Error while changing a slave address in protected operating mode: No slave with address 0 present
91	Error while changing slave address: Target address is already occupied (see chapter 4.6)
92	Error while changing slave address: New address could not be set
93	Error while changing slave address: New address could only be stored volatile in the slave
94	Error while changing slave address in protected operating mode: Slave has wrong configuration information (see chapter 4.6)
95	Error while changing slave address in protected operating mode: The configuration error is caused by one slave too many (instead of by a missing slave).

10 Appendix B: First Commissioning of AS-i

Note

In this chapter an example is given of how to put an AS-i network into operation quickly and easily and without the need for external devices. The addressing of the components connected to the AS-i network can be performed directly on the AS-i Master. It is of course more comfortable to do the addressing with a hand-held programming device or a PC.

However, it is possible to configure even complex networks using only the AS-i Master.

What to do ?	How to go about it?
See to it that the AS-i Master is properly supplied with power.	Using AS-i Master with power supply "A": connect the AS-i power supply unit to the terminals AS-i + and AS-i - of the master, connect the ground terminal. Using AS-i Master with power supply "N": connect the 24 V DC standard power supply with the terminals 24V and 0V of the master, connect the ground terminal. Turn on the power supply.
After the self-test: the LEDs "power", "config err", "U ASI" and "prj mode" are on. The LCD shows "40": the AS-i Master is in the off-line phase. Shortly after that a "41" will be displayed: the AS-i Master stays in the detection phase.	
Switch the device to the projecting mode, if the yellow LED does not light up.	Press the "mode"-button for approx. five seconds.
The yellow LED "prj mode" lights up. The device is now in projecting mode.	
Add a slave with the address 0 to the AS-i line.	Connect the slave's terminals with the terminals AS-i + / - of the master.
The green LED "ASI active" lights up. The LCD shows "0". This means the AS-i Master has detected the slave.	
Change the slave address to address 1.	Select address 1 by pressing the "set" button shortly, if necessary repeatedly. When a “1“ appears on the display press the “set“ button for approx. five seconds until the display blinks. Press again shortly the “set“ button to assign the new address to the slave.
The AS-i Master detects the slave with address 1 and displays "1".	

Connect another slave with address 0 to the AS-i line and allocate the address 2 to it.	Connect the slave to the AS-i line. The addressing is the same as for the previous slave.
The addresses of all slaves detected are now displayed sequentially.	
Change to the protected operating mode and store the AS-i configuration.	Leave the configuration mode by pressing the "mode" button for at least five seconds until the "prj mode" LED goes out.
The configuration of the master is now finished. If the AS-i Master is a gateway, the hierarchically higher fieldbus system can be put into operation now.	